SHARE – A EULOGY TO COOPERATIVE EFFORT

Paul Armer

P-969

October 31, 1956

To be presented at Electronic Business Systems Conference, San Francisco, on November 8, 1956.

The RAND Corporation
1700 MAIN ST. • SANTA MONICA • CALIFORNIA

SUMMARY

SHARE has been frequently described as a "users cooperative". It is made up of most of the organizations who have, or plan on getting, an IBM Type 704 Electronic Data Processing Machine. Its aim is to eliminate, as much as possible, redundant effort expended in using the 704. This paper discusses, briefly, the history of cooperative effort in the scientific computing field, as it existed in 1956. The history of SHARE itself is covered in detail. It is recommended that similar groups be formed to serve those interested in other areas of data processing.

SHARE — A EULOGY TO COOPERATIVE EFFORT

Whenever someone asks about SHARE, the first question is usually "What do the initials mean?" The answer is that SHARE is a name and not a set of initials. The second question is usually "Just what is SHARE?" SHARE has been frequently described as a "users cooperative". It is made up of most of the organizations who have, or plan on getting, an IBM Type 704 EDPM. Like any cooperative, SHARE was formed to be of service to its members. Its aim is to eliminate, as much as possible, redundant effort expended in using the 704. It seeks to accomplish this aim by promoting inter-installation cooperation and communication.

HISTORICAL SKETCH OF SHARE

As I attempt to paint a historical background for SHARE, it is important for you to remember two things about me, for what anyone has to say about the past is always greatly influenced by his vantage position. The two points are that my primary field is scientific computing and that all my experience has been with the equipment of one manufacturer, IBM. Although the latter point may affect what I have to say about the past, it has no bearing on my discussions of the future.

Before taking up SHARE itself, let's turn our attention to the history of cooperative effort in the field of machine accounting and computing. Since almost all early computing

efforts got under way in an accounting machine installation, any discussion of early cooperative effort in computing is necessarily concerned with the machine accounting field. To begin with, we "shared" machine wiring diagrams, usually by submitting such diagrams to the machine manufacturer, who reproduced them and distributed copies to the field. As an example of this, many of you are familiar with IBM's "Pointers". Another important vehicle for the interchange of information of this sort is Fred Gruenberger's "Computing News", published in Richland, Washington. This newsletter frequently publishes wiring diagrams and other "ideas" submitted by its readers. This kind of cooperative activity continues today, although not at the level some would like.

And while discussing cooperation and the interchange of information and ideas, the various professional organizations, in particular the NMAA, should be given much credit for their efforts.

But the important point about these early efforts at cooperation is that seldom, if ever, did individuals from more than one organization sit down together to develop something through cooperative endeavor which each could take back to his own installation and use. Actually, this wouldn't have made much sense in the early days when machine work was divided into many separate and distinct steps. In fact, I doubt if cooperation of this sort made any sense at all prior to 1950 when the Model I Card Programmed Calculator (CPC) was

introduced by IBM. Here, for the first time in punched card work, the concept of processing data in a serial fashion ("in-line"), rather than in parallel, was introduced. Now the CPC was really a computer kit rather than a finished calculator, for after it rolled in the door, one had to do a great deal of work designing, wiring and debugging a set of plug boards which connected the various pieces of the kit and made it into a calculator. Here then, was an opportunity for a cooperative effort in putting that kit together. This opportunity was completely overlooked, despite the fact that IBM brought together representatives from each of the organizations getting early model CPC's. Further, this meeting was held in advance of the delivery of the machines. The idea of a cooperative effort just didn't occur to anyone, for we were all too naive about the machine and about handling our work in this "in-line" fashion. But, most important to a moral I'd like to draw in this paper, we were all so naive that each of us believed that we could put the kit together better than anyone else. Consequently, we all went our separate ways and each of us ended up with a unique calculator.

I've somewhat overstated the "lost opportunity" aspects of this situation for it is probably true that, considering how little each of us knew about the machine and about "in-line" processing, it was necessary that we go back to our own installations and learn from our own mistakes. Nevertheless, some sort of sharing of information during the

next few years might have reduced the duplication of mistakes that resulted from the spirit of splendid isolationism which prevailed.

Lest anyone get the impression that I entirely disapprove of the versatility inherent in the CPC, let me hasten to add that I don't believe IBM should have delivered the CPC with a set of plug boards, designed by IBM, soldered into the machine, for they didn't know very much about the potentialities of the CPC at this time either. But versatility can be carried to an extreme — what could be more versatile than a kit made up of tubes, relays, resistors, condensers, etc., with each customer left to his own desires?

We also missed our second chance at a cooperative effort when the Model II CPC was introduced, although some of the later organizations to accept machines did copy and use setups designed by others. The fact that this opportunity was overlooked can not be laid on the doorstep of inexperience with this type of equipment. The blame must be placed on the "I can do it better" attitude.

When the 701 came along, we still weren't very wise and once again almost everyone went his own way. But this time the amount of redundant effort was horrendous — the cost of developing a system for using the machine, and a set of routines to go with that system, was usually in excess of a year's rental for the equipment. But strangely enough, it wasn't these factors which resulted in what I consider to be

the first successful cooperative effort in the field. I am referring to PACT, which is a set of initials and stands for the Project for the Advancement of Coding Techniques. But before discussing PACT further, let me return to the pressures which resulted in its birth.

In the fall of 1954, the several organizations who had been operating 701's in the Los Angeles area were going through a period of self-examination. The one thing plaguing all the organizations was the mismatch between the machine and its language and the human and his language. The elapsed time from problem origination to solution was frequently intolerable, problem check-out was difficult and expensive. People who had estimated that it would take a one-shift operation to handle their production load found themselves operating two shifts, not because they had missed their production estimate, but because they had overlooked a shift devoted to code-checking. Estimates of the cost of writing and checking a program ran as high as $10.00 per instruction. Training was difficult, took a long time and was expensive.

In response to these pressures, a number of interpretive systems were devised. These made problems easier to code and therefore reduced elapsed time and debugging difficulties. They reduced the training problem. But they introduced a new problem, one which frequently outweighed the advantages gained. The new problem was due to the fact that these interpretive routines slowed down the effective

speed of the machine by a factor between 10 and 100. There were no longer enough hours in the day to get the machine's work done.

At this point, Jack Strong and Frank Wagner of North American Aviation, suggested that a cooperative effort, aimed at developing an automatic coding system, be undertaken by the computer users in the Los Angeles area. The enthusiam of Strong and Wagner prevailed and PACT was born. The idea was to find a way to remove some of the coding burden from the human and place it on the machine without materially reducing machine efficiency. I do not intend to go into PACT here; it did produce a successful compiler for the 701 which is referred to as PACT-I. A series of papers describing PACT-I appears in the October 1956 issue of the Journal of the Association for Computing Machinery. The PACT group is presently working on PACT-IA, a compiler for the 704.

The important thing about PACT to my discussions today is that it is representative of the kind of cooperation where individuals from different organizations did sit down together to develop a system that each could take back to his own installation and use. In doing this, PACT rediscovered an age old truth that man has been forgetting and rediscovering over and over again since the Stone Age; i.e., cooperation is the greatest invention since the wheel. Actually, this was not an immediate discovery. The members of the working committee of PACT spent several weeks in mutual education, for

at first they had to overcome the "our way is best" attitude and also a serious language problem. That this mutual education led to mutual admiration and respect for the other fellows' abilities is testified to by the final report of the PACT-I working committee to the PACT policy committee. I quote from their Primary recommendation.

> "The spirit of cooperation between member organizations and their representatives during the formulating of PACT-I has been one of the most valuable resources to come from the project. It is essential that this spirit of cooperation continue with future project plans."

One might believe that in such a climate, an organization like SHARE would have developed almost spontaneously when the task of preparing for the advent of the 704 appeared. Strangely enough, it was not spontaneous, but rather somewhat of an accident, for even this opportunity for a major cooperative effort almost escaped us.

Three 701 installations in the Los Angeles area began to dig into the problem of preparing for the 704 in the summer of 1955. Because of the climate resulting from PACT-I, these three organizations started to discuss their individual plans with each other and to explore the possibilities of a joint effort in connection with program development for the 704. Accordingly The RAND Corporation, Lockheed Aircraft Corporation and North American Aviation, Inc. seriously

began to consider standardization. This much of SHARE genesis was no accident — it flowed naturally from the PACT experiences of the three groups. The fortunate accident was a seminar held by IBM in Los Angeles early in August for all Western installations considering the 704. The cooperative venture being launched by the three local groups was discussed with others at the seminar and although SHARE may not have started spontaneously, the fire soon burned furiously and spread rapidly across the country. Two weeks after the IBM seminar, the first meeting of SHARE was held at RAND during the week of August 22, 1955. Despite short notice, almost all (18 in number) . . installations then contemplating the 704 were represented at the meeting.

I mentioned a minute ago that this opportunity almost escaped us. The problem was a matter of timing, for several organizations were expecting their equipment within three months after the initial meeting and had their systems for using the machine nearly complete. Of the four organizations well along in their plans, one was able to go along with SHARE when their system was adopted, with modifications, by the SHARE body. A second elected to junk what work had been done to date in order to go along. Two others were much too far along with their own systems to turn back; for them, SHARE did come too late.

I think it is important here to understand that SHARE was not organized just to facilitate the interchange of

programs for the 704. This was a higher order of cooperation. The organizations who had interchanged 701 programs had found the routines of others almost useless, for each installation had its own system and a routine designed for one system just wouldn't fit into another system without modifications. Hence, it was usually easier to write a routine for your own system, starting from scratch, than to modify someone else's routine. And so, almost everyone wrote his own.

Actually, I personally believe that some of this reluctance to modify and use somebody else's routines can be traced to that naivety mentioned earlier in discussing the CPC; i.e., the belief that the other guy didn't really know what he was doing and that "I can do it better". In any event, the interchange of programs for the 701 had not, in general, been very successful.

At the first meeting of SHARE, disdain for the other fellow's abilities was gone — there was general "agreement to agree" — and almost all professed themselves as quite willing to accept the ideas of others, even to the extent of obsoleting things already done within their own installations. This spirit, however, was not carried to an extreme, for one of SHARE's principles is "unity in essentials and freedom in accidentals". Standardization is undertaken only where necessary. Let me quote from a statement of the "Obligations of a SHARE Member":

"The principle obligation of a member is to have a cooperative spirit. It is expected that each member approach each discussion with an open mind, and, having respect for the competence of other members, be willing to accept the opinions of others more frequently than he insists on his own. On the other hand, majorities of members are not expected to be overbearing in their dealings with minorities. To win over dissenters to unanimity and not to vote them down is the foremost objective in every discussion. When it comes to standards, SHARE insists on adherence to them for communication purposes through SHARE channels to the extent that it refuses to distribute material not in SHARE language. Of course, decisions of SHARE can in no way be binding on any member installation so far as its internal operation is concerned. However, the great majority of SHARE members deviate internally only very slightly or not at all from the standards adopted by SHARE. New members are urged to scrutinize carefully any such deviation before deciding that it is imperative that they do so. Please note that the foregoing discussion refers to basic contradictions or radically different ways of doing things, and does not refer to minor

improvements and additions which will not in the least interfere with normal communications."

As evidence that the SHARE membership paid more than lip service to these principles, let me point to the solid accomplishments of the first meeting of SHARE. After deciding on a loosely knit organizational structure and electing officers, attention was turned to those areas where standardization was essential to inter-installation communication. SHARE standards were adopted for a mnemonic operation code, assembly program, card format and print wheel configuration. A distribution system, the lifeline of the organization, was established. Without this distribution system, SHARE could not exist in the fashion that it does. Among the other decisions made were a definition of what constituted a minimum 704, the location of the binary point and the conventions to be used in writing subroutines. Along the latter lines, the work required to prepare various utility and mathematical routines for the machine was divided among the member installations on a purely voluntary basis. Another item of business of that first meeting was the appointment of a committee to prepare a glossary of terms to supplement the existing computing dictionaries. This came about when we soon realized that we were faced with the language problem which had plagued FACT in its early days.

I don't want to leave you with the impression that all SHARE decisions came easily. There was frequently much wrangling and discussion. But in each case, a spirit of cooperation prevailed and a compromise was reached.

The second meeting of SHARE was held some three weeks later in Philadelphia. This meeting was primarily devoted to a re-evaluation of the assembly program and to reporting on the programming commitments made at the first meeting. Of the thirty-seven programming assignments made at the first meeting, all but two were completed on schedule and more than twenty additional programs were submitted.

Subsequent meetings of SHARE were held in Boston, San Francisco, Chicago, and Denver. A meeting is to be held next month in New York and, in the coming year, meetings have been scheduled for Dallas and San Diego.

Some other topics which have been covered at these meetings include: the use of peripheral equipment, suggested changes to the 704 and to the peripheral equipment, the use of the cathode ray tube display device (the type 740), changes to the assembly program, discussion of forms, standard printer boards, computer layouts, development of a SHARE reference manual, the cataloging of SHARE programs, machine reliability (in particular, and a favorite topic of mine, tape reliability), diagnostic routines, education (both internal and external), machine statistics, programming in general, gadgets built to facilitate use of the computer,

debugging techniques, data reduction, data transmission systems and, of course, as the membership grew, we found it necessary to devote some time to our organizational structure.

As of this writing, the SHARE membership has grown from 18 to 62. Included are installations in Canada, France and England. These 62 organizations have some 76 machines on order. Including associated peripheral and punched card equipment, the combined annual machine rentals for the present SHARE membership will one day easily exceed $50,000,000.00.

In addition to the 62 member installations, there are 88 additional organizations on the non-member distribution list for program write-ups.

ADVANTAGES OF SHARE

Some three hundred programs have been distributed to the membership. There is surprisingly little duplication in this library. In the early days of SHARE, it was a standard joke that everyone was submitting square root routines, since they made convenient assignments for trainees. Nevertheless, there are only five square root routines in the literature. But more important, there is only one for such things as matrix abstraction. There are only three general printing routines. Needless to say, without a cooperative effort like SHARE, there would soon be at least fifty versions of most of the more important routines in the SHARE library.

Using the rough rule of thumb that the cost of setting

up a system and its associated routines for a computer is approximately equal to the first year's rental for the equipment, we arrive at the conclusion that the savings to the membership, as a result of the reduction of redundant programming effort, is in the neighborhood of $50,000,000.00.

This seems quite reasonable — consider only the assembly program, which was originally developed by United Aircraft Corporation and subsequently modified by them to conform with suggestions from the SHARE body. By any standards, it's an elegant and complicated assembler. Consequently, it seems appropriate to assume that the cost per instruction in it is at the high end of the $2.00 to $10.00 scale usually quoted as the cost per instruction. Applying the $10.00 rate, we conclude that to develop a similar assembler would cost an "isolationist" some $25,000.00. Although not all the members of SHARE are using this assembly program, most are and therefore we may conclude that the resulting savings are of the order of $1,500,000.00.

Even so, there is a more important point here. Many of the later 704 customers are taking the giant step from slide rules, desk calculators, and/or CPC's to the 704 without the benefit of very much intervening experience with stored program equipment. On the other hand, the SHARE assembler and most of the other routines were developed and written by personnel with considerable 701 experience. Many of the newer 704 users have expressed the opinion that with-

out SHARE they would have been unable to go so far up the computing capability ladder in a single step. In effect, SHARE has multiplied the efforts of the limited number of experienced computer personnel. Not only has it made available programs the newcomers might not have been able to produce for themselves, but in those organizations having a number of experienced personnel, the reduction of redundant effort has released many such people for work on more sophisticated utility and mathematical routines and on applied problems.

Another important advantage of SHARE flows from the personal acquaintanceships developed at its meetings. Subsets of the membership discover common problems — there is much cooperation at the two and three installation level. Information and ideas are continually being interchanged between members, both inside and outside the meetings. Because of the meetings and the distribution system, the transmission of information and ideas is made much easier.

Yet another advantage lies in an area which I haven't mentioned so far. In these days of automation, one of the much used "okeh" words is "feedback". SHARE provides collective "feedback" from the customers to the manufacturer. To me, this is extremely important. Both the customer and the manufacturer are vitally interested in improving the present equipment, in filling needs presently unfulfilled, and in seeing that the next generation of machines properly reflect the customers' needs. As an example of this, consider

peripheral equipment. Designed for use with the 702 and 705, the peripheral equipment originally dealt only with cards using the Hollerith code. But SHARE felt a need for reading and punching binary cards. At SHARE's request and with suggestions from SHARE, a method was worked out to do so. SHARE has also provided IBM with collectively considered requests for changes to the 704 itself. And although SHARE has explicitly decided to limit its area of activity to the 704, the discussions between customers and manufacturer at SHARE meetings cannot help but have considerable effect on the computers of the future.

DISADVANTAGES OF SHARE

I came here to praise SHARE and not to bury it, despite the connotation of "eulogy" in my sub-title. Actually, there is little to say on the disadvantages of SHARE. I think they're all rather obvious. Most important, but still of trivial import on an absolute scale, is that standardization obviously implies some loss of flexibility. And of course, SHARE provides 3 or 4 more meetings per year to be attended. These days, it is almost literally true that one can find enough meetings, in the EDP field, to enable one to avoid ever having to go to the office.

OTHER COOPERATIVE EFFORTS IN THE COMPUTING FIELD

Anyone who will look at IBM Technical Newsletter #10 can conclude, by observing the number of "Systems" for the IBM Type 650 reported on therein, that a great deal of

redundant effort went into these systems. And it still is. However, this situation was probably, to some extent, unavoidable. It's like things were with respect to the CPC; each user had to learn about the stored program concept, by his own missteps, before he could be ready for a cooperative effort. Nevertheless, I'm convinced that the 650 area could benefit greatly from some sort of cooperative effort.

For the Remington-Rand Univac Scientific Model 1103A, there is positive information to report. The users and prospective users of this equipment have banded together in a group called USE (Univac Scientific Exchange) with much the same aims as SHARE. This talk could just as well have been given by a member of USE.

Also, the organization of a cooperative group for the IBM Type 705 is underway with the first meeting scheduled for New York during the first week in December.

THE FUTURE

I'm sure that the cooperative effort for the next model computer will come early and not be almost too late like SHARE. There are undoubtedly other things which will be different this time. Remember that SHARE came into being long after several prospective 704 users had their own systems under development. Because of this, when SHARE considered the question of a standard assembly program, several were essentially finished. SHARE picked one of these (that of the United Aircraft Corporation), with modifications, as its

standard. This meant that almost all the burden for the assembler fell on UAC. This time we hope to apportion the load, while combining the ideas of many, by making the assembler the joint effort of a number of installations. This may not be easy because of geography. Few SHARE activities in the past have required that the personnel concerned work together in the same physical location for an extended period of time. However, if we are to have a joint assembly program, a way must be found to lick this problem.

COOPERATIVE EFFORTS IN THE DATA PROCESSING AREA

As a preface to this topic, it is important to consider the ways in which scientific computing differs from business data processing. In the former field, we are faced with a large number of problems for the computer, most of them fairly small and non-repetitive in the sense that they may be in the production phase for less than a month. In such circumstances, any "good" way to solve the problem is preferred to spending time in search of the "best" way. One tries to develop a "general purpose" system through which almost all the problems can be pushed with a minimum of over-all effort. Tools in the form of utility and mathematical routines are developed to aid in attacking problems with some common attribute. Since these systems are "general purpose" in nature, they are as useful in one computing installation as another.

I needn't tell you that things are much different in

the area of business data processing. Here there are a few very large applications which will be used over and over again. In these circumstances, it pays to search for the "best" way of doing a problem and to polish the final program in the interest of machine efficiency. Consequently, the "special purpose" approach is normally preferred to the "general purpose" method.

Another way in which problems of the two fields differ is important when considered in the light of cooperative effort. This difference is that computing deals with an exact science in Mathematics while business data processing deals with the vagaries of the world. The logarithm of a given number is identically the same in every installation and consequently a routine for calculating a logarithm can be gainfully passed among computing installations and used "as is". But could any of you make use, "as is", of the payroll routine of another company?

However, I don't mean to belittle the advantages of a cooperative effort in the business data processing field. On the contrary, I feel such an effort would pay tremendous dividends, even if limited just to getting the people with common problems together. And this reminds me of another important point about SHARE, where the idea is to get top-quality working-level personnel -- not just the chiefs -- together to discuss common problems. The resulting "mutual education" has been invaluable. I believe that this is an

important concept and one which accounts for much of SHARE's success. I feel quite strongly about this point — much is to be gained from the cross-fertilization of top-quality working-level personnel. The section of the SHARE Reference Manual pertaining to "Obligations of SHARE Membership" contains this statement, "...it is desirable that each SHARE member be represented at every meeting by at least two men, one empowered to make basic policy decisions and another thoroughly familiar with techniques, programming and detailed operating matters."

But a cooperative effort in the business data processing field need not be limited to a series of seminars. Much could be done to facilitate inter-installation communication and joint endeavor could be brought to bear on common problems. Even the fact that a number of installations have been in actual operation for some time should not hinder the success of a cooperative effort organized for a specific machine. It is not necessarily too late. Much standardization may already exist due to the common practice of adopting the mnemonic code and assembly program supplied by the manufacturer. Inter-installation communication may come fairly easily — further standardization may not be difficult. Rather than being too late, this may be the first time that a cooperative effort is possible for a group having a common machine. For example, it is not reasonable to expect that users of the Univac (or the 705) could have gotten together in the past

to adopt standards and to share the burden of preparing for the machine. For almost all the organizations concerned, this was their first encounter with a stored program machine. As with the CPC, a great deal had to be learned by each company about the equipment and about this new way of doing things before a cooperative effort could be undertaken.

As it was with SHARE, I feel that the success of any such venture will depend on the degree to which an attitude of "agreement to agree" pervades the membership. This attitude must go hand in hand with mutual respect for the ideas and opinions of others.

It's redundant for me to say, in summary, that I am enthusiastic about SHARE and about cooperative effort in general — I hope it's contagious.